

# Punktwolken zu BIM – Methoden der Bauteilanpassung in Laserscannerdaten

## Point Clouds to BIM – Methods for Building Parts Fitting in Laser Scan Data

Christian Tonn, Oliver Bringmann

Bauen im Bestand erfordert Unterlagen über bestehende Gebäude in einer Form, wie man sie vom Neubau oder Architekturentwurf kennt. Laserscanning ist ein mächtiges Verfahren, um die gebaute Realität zu erfassen. Es liefert den Abguss des Bestands in Form von Punktwolken. Die Schwierigkeit besteht darin, die entstehenden Datenmengen auszuwerten und als ein Gebäude-/Anlagen-Informations-Modell (BIM) zu interpretieren. Anhand der Wanderkennung wird ein generischer Ansatz zur Mustererkennung von Architekturobjekten vorgestellt. Im zweiten Teil wird diskutiert, wie einzelne Bauteile vollautomatisch zu konsistenten Netzwerken verbunden werden. So sollen Wände in der BIM-Software im Rahmen einer vorgebbaren Genauigkeit orthogonal bzw. fluchtend zueinander ausgerichtet werden.

**Schlüsselwörter:** Punktwolken, BIM, Mustererkennung, Bauteile, Wände ausrichten

*New construction within existing contexts requires documentation of the existing buildings in a form that one is familiar with from new construction or architectural design. Laser scanning is a powerful method to capture the built reality. It provides a replica of the existing building in the form of a point cloud. The challenge is to analyze the resulting amounts of data that has been generated and being able to interpret it as a Building Information Model (BIM). On the basis of wall detection a new generic approach for pattern recognition of architectural objects is presented. The second part describes how individual components can be automatically connected to consistent networks. By using a BIM software walls should be aligned, within predefined limits of accuracy, either perpendicular to or in line with each other.*

**Keywords:** Point cloud, BIM, pattern recognition, components, wall alignment

### 1 EINLEITUNG

Building Information Modelling (kurz: BIM) hat sich zum Schlagwort für zukunftsweisende Prozesse und neuartige Software in der Bauindustrie entwickelt. Anhand eines digitalen, dreidimensionalen BIM-Modells lassen sich alle notwendigen Planungssichten – sowohl geometrische, wie Grundrisse, Ansichten, Schnitte, als auch nichtgeometrische, wie Raumbücher oder Bauteillisten – ableiten. Mit zunehmender Durchdringung der BIM-Methodik in der Bauplanung steigt auch unweigerlich die Forderung nach BIM-Modellen von existierender Bausubstanz an. Dabei stellt sich die Frage nach der effektivsten Methodik zur Erfassung der Bestandsgebäude. Das terrestrische Laserscanning erscheint hierfür ideal. Es ermöglicht die schnelle und präzise Erfassung von 3D-Daten in hoher räum-

licher Auflösung. Dabei ist der Prozess aber keineswegs trivial. Die Herausforderung liegt darin, aus der präzisen Abbildung der Realität in Form von Punktwolken ein parametrisches 3D-BIM-Modell abzuleiten. Voll- oder besser halbautomatische Verfahren sind hierfür notwendig, um den zeit- und kostenintensiven Prozess der manuellen Nachmodellierung zu beschleunigen und plausibler zu gestalten.

## 2 VOLLAUTOMATISCHE VS. HALBAUTOMATISCHE MUSTERERKENNUNG

Selbstverständlich wünscht sich jeder Nutzer eine automatische und fehlerfreie Mustererkennung von Bauteilen in Punktwolken. Doch vollautomatische Verfahren kommen bei realer Architektur schnell an ihre Grenzen. Die anschließende Zeit für Fehlererkennung und -beseitigung übersteigen in der Regel den anfänglichen Vorteil der Automatisierung. Gesucht sind daher Workflows, welche die Steuerung beim Nutzer belassen, aber das händische Modellieren signifikant verbessern.

Wie bereits in /Braunes 2014/ erläutert wurde, sind vollautomatische Verfahren in der Praxis aus folgenden Gründen ungenügend:

- Störobjekte in Punktwolken wie Möbel, Verkleidungen und Grünpflanzen verhindern die automatische Mustererkennung, da sie im Zielsystem nicht modellierbar oder aussortierbar sind.
- Reale Objekte werden in der BIM-Software meist stark generalisiert modelliert. In der Wirklichkeit existieren viel mehr Details (Abb. 1) und Ausprägungen, als in der Software nachgebildet werden können. Dadurch weicht das als korrekt wahrgenommene CAD-Modell von der Punktwolke ab. Diese für einen guten Vollautomaten nötige Abstraktionsleistung ist noch nicht Stand der Technik.
- Punktwolken sind teilweise durch Scan-Abschattungen unvollständig.

Halbautomatische Verfahren, welche den Nutzer mit einbeziehen, sind an diesen Stellen klar im Vorteil, da sie das Fachwissen und die Erfahrung des Anwenders zur Modellierung und Mustererkennung mit verwenden können sowie den Arbeitsprozess genauer und schneller durchführbar gestalten, als einfaches manuelles Darüberzeichnen.

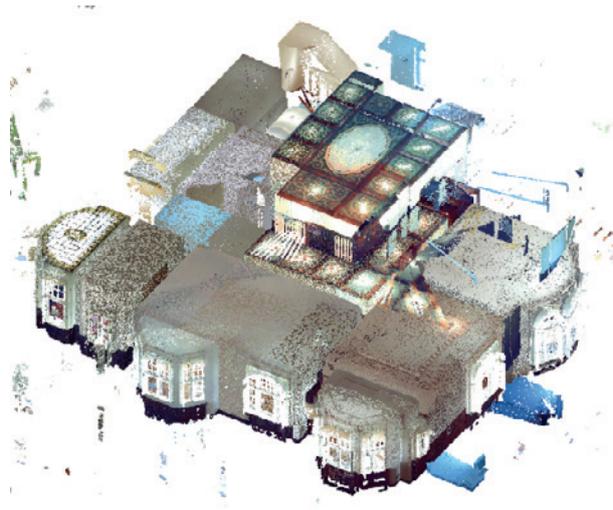


Abb. 1 | Detailreiche Punktwolke eines Gebäudeerdgeschosses

Im ersten Schritt werden die dreidimensionalen Punktdaten der zu untersuchenden Etagenschicht des Gebäudes auf ein Grundriss-Punktdichte-Bild mit Schwellenwert, im Folgenden als Graustufenbild bezeichnet (Abb. 2), reduziert. Angenommen wird eine senkrechte gerade Wand. Die ungefähre Ausdehnung in der Höhe sei ebenfalls gegeben. Damit lässt sich das Problem auf eine 2D-Analyse reduzieren. Der relevante Bildausschnitt (Abb. 4) wird mittels einer angepassten Distanztransformation umgewandelt in ein zweites Bild, welches die kürzeste Distanz eines jeden Bildpunkts zur nächsten dichtesten Punktwolkenzelle darstellt (Abb. 5).

Im Unterschied zum bei /Bringmann 2012/ vorgestellten Verfahren zur Erzeugung des „Potenzialgebirges“ wird in diesem Anwendungsfall ein Graustufenbild (Grundriss-Punktdichte-Bild mit Schwellenwert) in das „Potenzialgebirge“ umgewandelt. Dieser Ansatz berücksichtigt dabei ebenso die Punktdichte-Information der Punktwolke. Das Graustufenbild wird in acht Helligkeitssegmente

## 3 WÄNDE ANGLEICHEN

In der aktuellen BIM-Software muss der Modellierer vor dem Erstellen einer Wand ihren Typ mit entsprechender Wandstärke auswählen. Anschließend werden die exakten Anfangs- und Endpunkte der Wandachse gewählt.

Beim Fitting von Wänden sollten die Informationen zu Wandtyp und Wandstärke gleichzeitig mit der korrekten Wandachse aus der Punktwolke abgeleitet werden. Die zu lösende Aufgabe besteht darin, nach zwei Mausklicks irgendwo im Inneren der Wand die Lage und den Typ zu bestimmen und die Wandinstanz einzufügen.

Das grundlegende Verfahren des Angleichens von analytischen Geometrien in Punktwolken wurde bereits in /Bringmann 2012/ beschrieben. Im

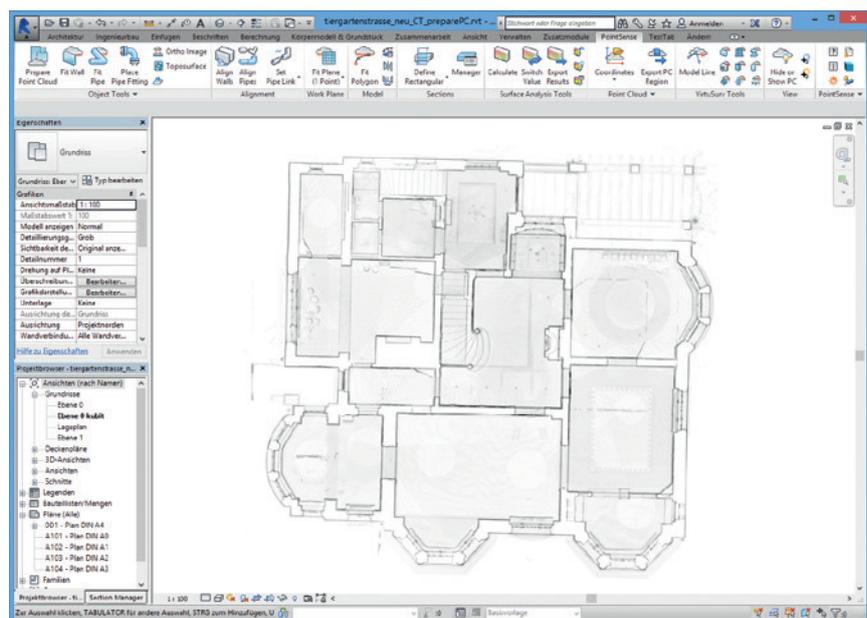


Abb. 2 | Punktwolkenschicht als Grundriss-Punktdichte-Bild mit Schwellenwert

gesplittet, welche, wiederum als Schwarz-weiß-Bild betrachtet, distanztransformiert werden. Diese acht distanztransformierten Bilder werden anschließend wieder – entsprechend ihres ursprünglichen Helligkeitsbereichs gewichtet – zu einem „Potenzialgebirge“-Bild zusammengeführt.

Des Weiteren wird das „Potenzialgebirge“ in zwei Halbbilder getrennt. Die Grenze bildet dabei die initiale Klicklinie des Nutzers. Ausgehend von dieser Grenze gibt es ein oberes und ein unteres Halbbild, in welchen unabhängig voneinander die Punktwolkendichtebilder in „Potenzialgebirge“ umgewandelt werden. Hiermit vermeidet man, dass Punktwolkenpunkte auf einer Seite der initialen Klicklinie das Ergebnis des Linienfittings der gegenüberliegenden Seite beeinflussen.

In diesen Distanzbildern kann besonders schnell eine lokale Minimierung von zwei parallelen Linien durchgeführt werden. Diese Linien repräsentieren die Innen- und Außenfläche einer geraden senkrecht stehenden Wand.

Die drei zu bestimmenden Optimierungsparameter dieser Wand-Angleichen-Aufgabe sind (Abb. 3):

- der Drehwinkel  $\alpha$  der Wandachse zur initialen Klicklinie,
- der „obere“ Abstand  $d_1$  der Wandoberfläche und
- der „untere“ Abstand  $d_2$  der Wandoberfläche zur gedrehten Klicklinie.

Im Unterschied zu vollautomatischen Verfahren kann der Nutzer hier zusätzlich noch korrigierend eingreifen, falls das falsche Liniensegment (z. B. eines Fensters etc.) für das lokale Optimum der Wandoberfläche gefunden wurde. Nach einem ersten schnellen Optimierungsdurchlauf kann der Nutzer optional bis zu vier Korrekturpunkte im Bild anklicken, welche korrigierend die Startposition der beiden Liniensegmente beeinflussen. Nach der Eingabe eines Korrekturpunkts wird eine neue Optimierung durchgeführt, um die Wandinnen- und -außenseite an die Punktwolke anzuschmiegen.

Nachdem man die Information der Wandstärke hat, kann aus dem nun vorsortierten BIM-Katalog aller Wandtypen (Abb. 6) der passendste Wandtyp durch den Nutzer ausgewählt und eingezeichnet werden oder entsprechend der genauen Wandstärkemessung sinnvoll gerundet angepasst und weiter verwendet werden.

## 4 WÄNDE AUSRICHTEN

In einem nächsten Arbeitsschritt werden die gefitteten Wände, welche alle exakt den Messwerten der Punktwolke folgen, den Erfordernissen eines generalisierten BIM-Modells angepasst /Bringmann 2010/. Zu diesen Anforderungen zählen: Rechtwinkligkeiten, Parallelitäten, durchgehende Fluchten bei wechselnden Wandstärken und automatisches Verschneiden/

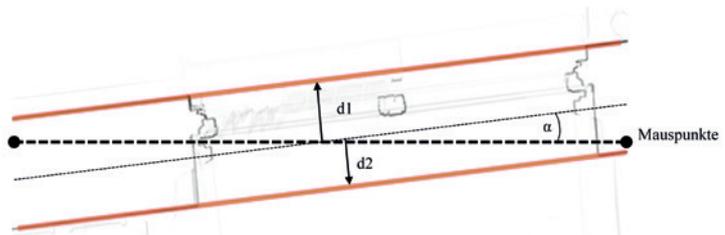


Abb. 3 | Schema des Wände-Angleichens

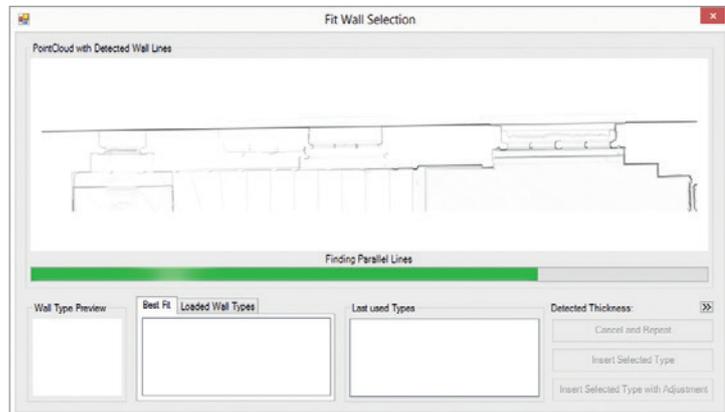


Abb. 4 | Optimierungsverfahren des Wand-Angleichens – zwei parallele Linien finden

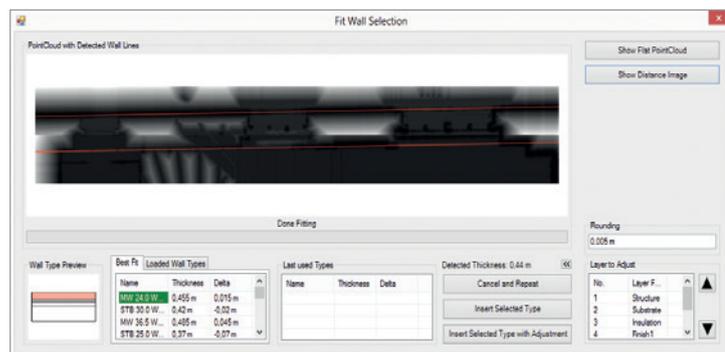


Abb. 5 | Distanzbild mit gefundenen roten Linien am lokalen Minimum

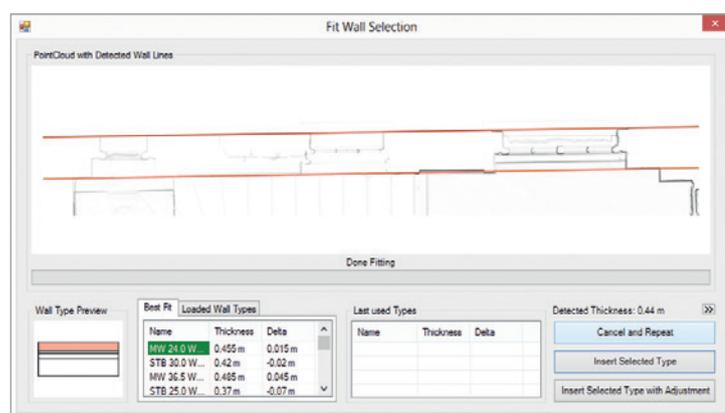


Abb. 6 | Gefundene Wandstärke und Lage – Darstellung des vorsortierten Wandtypkatalogs

Strecken von Wandsegmenten (Abb. 7). Das Kernproblem ist die Bildung von optimalen, disjunkten Wand-Clustern. Innerhalb eines Clusters werden die Anforderungen so durchgesetzt, dass die Abweichung zur Punktwolke minimal wird.

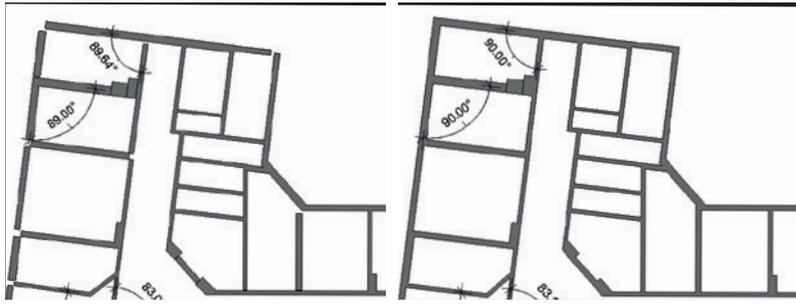


Abb. 7 | Wandsegmente eines Grundrisses vor und nach der Funktion „Wände Ausrichten“. Die Wände eines Referenzsystems werden orthogonal, parallel und fluchtend ausgerichtet

Der Algorithmus hierzu lässt sich grob in drei Schritte gliedern, welche nacheinander abgearbeitet werden (Abb. 9):

1. **Drehen** – Alle Wandsegmente werden innerhalb ihrer „maximalen Abweichung für Referenzsysteme“ gedreht, sodass parallele Cluster entstehen. Ein Referenzsystem besteht hierbei aus einem Hauptrichtungswinkel und einem Inkrementwinkel (meistens  $90^\circ$ ) in der XY-Grunde Ebene. Allen Wänden wird ein solches Referenzsystem zugeordnet.
2. **Fluchten** – Für alle parallelen Wände wird überprüft, ob sie innerhalb ihrer „maximalen Abweichung für Wandachsenausrichtung“ mittels Verschiebung auf eine gemeinsame Achse gebracht werden können. Diese Achse kann entweder auf der Innen- oder der Außenseite der Wand liegen. Es wird die Seite gewählt, deren Abweichung von der gefitteten Startposition der Wand geringer ist. Für die Wandachsenerzeugung werden wieder Cluster gebildet.
3. **Wände verschneiden** – Innerhalb eines Suchradius werden alle Enden der Wandsegmente automatisch verschritten oder verlängert, um ein geschlossenes Grundrissbild zu erzeugen.

#### 4.1 Wände Drehen

Im Folgenden wird der Algorithmus halbformal detaillierter wiedergegeben:

1. Die Menge aller Wände sei  $W = \{w_1, \dots\}$ .
2. Erzeuge nutzerdefinierte Referenzsysteme  $NRS = \{nrs_1, \dots\}$  durch Vereinigung von:
  - a. Gegebenenfalls: ein Einziges für den Norden (mit Hauptrichtungswinkel  $0^\circ$ ).

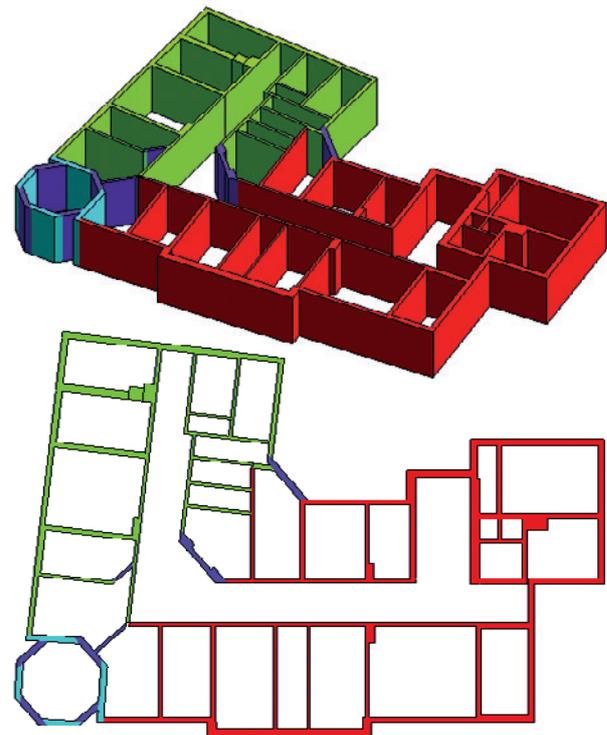


Abb. 8 | Detektierte und eingefärbte Cluster eines Grundrisses

- b. Gegebenenfalls: für jede durch den Nutzer manuell angelegte Referenzrichtung. Es entstehen so viele Cluster  $C = \{c_1, \dots\}$ , wie NRS Einträge hat. Jedes Cluster hat Zugriff auf seine Hauptrichtung  $c.richtung$ .

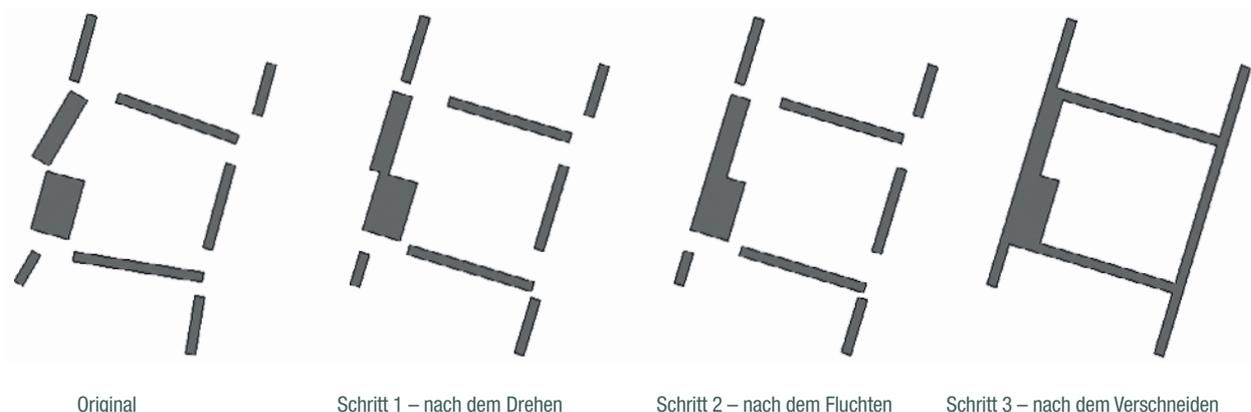


Abb. 9 | Die drei Schritte des Algorithmus

# Fokus auf Präzision ...

3. Alle Wände  $w$  werden versucht, dem bestpassenden Cluster  $c$  zuzuordnen, d. h.  $w$  ist genau dann ein Element von  $c$ , wenn für alle  $c'$  ( $c'$  ungleich  $c$ ) gilt:  
 $winkel(w.richtung, c.richtung + n \cdot Inkrementwinkel)$   
 $< winkel(w.richtung, c'.richtung + n \cdot Inkrementwinkel)$   
UND  
 $winkel(w.richtung, c.richtung + n \cdot Inkrementwinkel)$   
 $< winkelToleranz$   
mit natürlicher Zahl  $n$ , die zu einem minimalen Winkel führt. Es bleiben in Ausnahmen Wände übrig.
4. Erzeuge durch automatische Clusterung (Abb. 8) mögliche Richtungen  $ARS = \{ars_1, \dots\}$  von den übrigen Wänden  $AW := W - (Vereinigung \text{ aller Wände } c)$ .  
Es entstehen minimal 0 und maximal  $|AW|$  Cluster  $AC = \{ac_1, \dots\}$ .
5. Ordne alle Elemente aus  $AW$  einem  $ac$  analog obiger Regel 3 zu.
6. Die automatisch gefundenen Cluster  $AC$  werden nun der Menge aller Cluster  $C$  hinzugefügt:  $C := C + AC$ .
7. Drehe alle Wände eines Clusters  $c$  auf  $c.richtung + n \cdot Inkrementwinkel$  mit einer natürlichen Zahl  $n$ , die zu einem minimalen Winkel führt.  
Das sollte für ein Cluster immer funktionieren.

## 4.2 Wände Fluchten

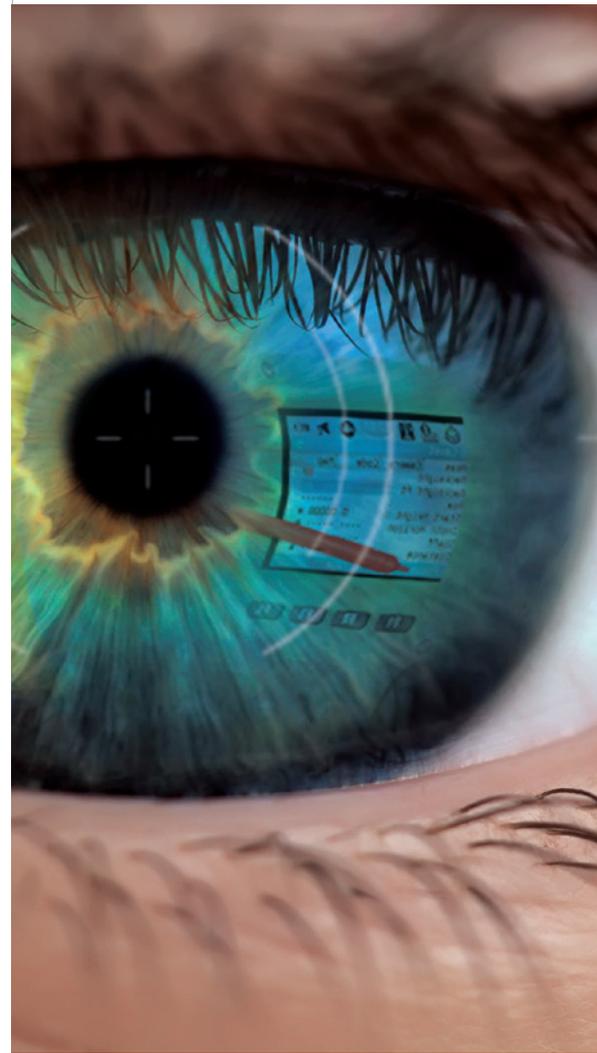
1. Für alle Teilmengen paralleler Wände erzeuge durch automatische Clusterung mögliche Achsen  $A = \{a_1, \dots\}$ . Die Achse  $a$  eines Fluchtend-Clusters  $fc$  heißt  $fc.a$ .
2. Schiebe alle Wände eines Clusters  $fc$  auf  $fc.a$ , wenn die „maximale Abweichung für Wandachsenausrichtung“ dies erlaubt. Wenn mehrere passen, nimm das Beste, welches in der entstehenden Summe der Fehlerquadrate am kleinsten ist.  
Das sollte für ein Cluster immer funktionieren.

## 4.3 Wände Verschneiden

1. Nun werden die Wandsegmentenden betrachtet: Es werden alle nächsten Schnittpunkte in der  $XY$ -Ebene zu benachbarten Wandsegmenten gebildet, welche innerhalb des Verschneideradius liegen. Diese Schnittpunkte werden bei jeweils beiden beteiligten Wandsegment-Enden gesammelt.
2. Das Wandsegment-Ende wird nun zu seinem am weitesten entfernten Schnittpunkt mit den Nachbarsegmenten verlängert oder gekürzt.
3. Die Wand in der verwendeten BIM-Software ist über seine mittlere Achsenlinie definiert. Da die Wandachsen nun topologisch und geometrisch korrekt verbunden sind, übernimmt das genaue Ausmodellieren des Wandanschlusses die verwendete BIM-Software.

Im Folgenden soll beschrieben werden, wie die „automatische Clusterung“, welche sowohl beim Drehen als auch beim Fluchten eingesetzt wird, funktioniert. Der Algorithmus adaptiert dabei das „ $k$ -nearest neighbour“-Verfahren /Altman 1992/. Der Unterschied besteht darin, dass die Clusteranzahl entsprechend der maximal erlaubten Wandabweichung bestimmt werden soll. Als erstes soll das Drehen betrachtet werden:

4. Bilde für jede Wand ein eigenes Referenzsystem, welches nicht schon mit diesem Hauptrichtungswinkel vorhanden ist. An dieser Stelle sollte der Fehler (d. h. die „maximale Abweichung für Referenzsysteme“) noch null betragen.
5. Falls der Fehler unterhalb der gesetzten Toleranz liegt, verfähre wie folgt:
  - a. Reduziere die Menge aller Referenzsysteme um dasjenige, welches in der Summe der Fehlerquadrate die geringste Fehlererhöhung bringt. Es werden also alle Referenzsysteme auf ihr „Weglassen“ geprüft.
  - b. Schlage die Wände des aussortierten Referenzsystems dem am besten passenden noch vorhandenen Referenzsystem zu.



### ... die neuen Leica Digitalnivelliere!

Automatisierte Funktionen und eine branchenführende Genauigkeit von 0,2 mm mit Standard Invar-Nivellierlatten liefern höchste Präzision.

Mit nur einem Tastendruck werden vor jeder Messung automatisierte Neigungsprüfungen durchgeführt. Dank dem integrierten Autofokus wird nicht nur Ihr Ziel schneller erfasst, sondern auch die Messgenauigkeit erhöht, indem der Kontrast der Latte maximiert wird.

- c. Berechne den resultierenden maximalen Fehler. Liegt er über der „maximalen Abweichung für Referenzsysteme“, brich diese Schleife ab, sonst fahre bei Punkt a fort.

Auf diese Art und Weise wird automatisch die notwendige Anzahl der Referenzsysteme mit ihren Hauptwinkelrichtungen ermittelt, um innerhalb einer bestimmten maximalen Abweichung zu bleiben.

Dasselbe Verfahren wird für die automatische Clusterung beim Fluchten eingesetzt. Man beginnt mit allen Wänden in einer eigenen „Fluchten“-Gruppe und reduziert anschließend solange jeweils um die fehlerärmste Gruppe, wie man noch innerhalb der vom Nutzer definierten Toleranz liegt.

Man erkennt, dass es sich bei diesem Ansatz um einen Algorithmus mit quadratischer Zeitkomplexität in Bezug auf die Anzahl der Wände handelt. Der Ansatz kann durch Heuristiken beschleunigt werden, welche die Gruppe der zu untersuchenden Wände schon im Vorfeld unterteilt – zum Beispiel in Wandgruppen, die durch die gesetzte Fehlertoleranz niemals miteinander fluchten könnten. Dadurch, dass die einzelnen Rechen- und Fehlerbewertungsschritte selbst nicht so zeitaufwendig sind, kann das Verfahren auf ganze Gebäude in vertretbarer Zeit angewendet werden. Drei Etagen des abgebildeten Gebäudes (Abb. 10) benötigten etwa drei Sekunden für das Berechnen von „Wände Ausrichten“ auf einem handelsüblichen Desktop-PC (Core i5-3570; 3,4 GHz).

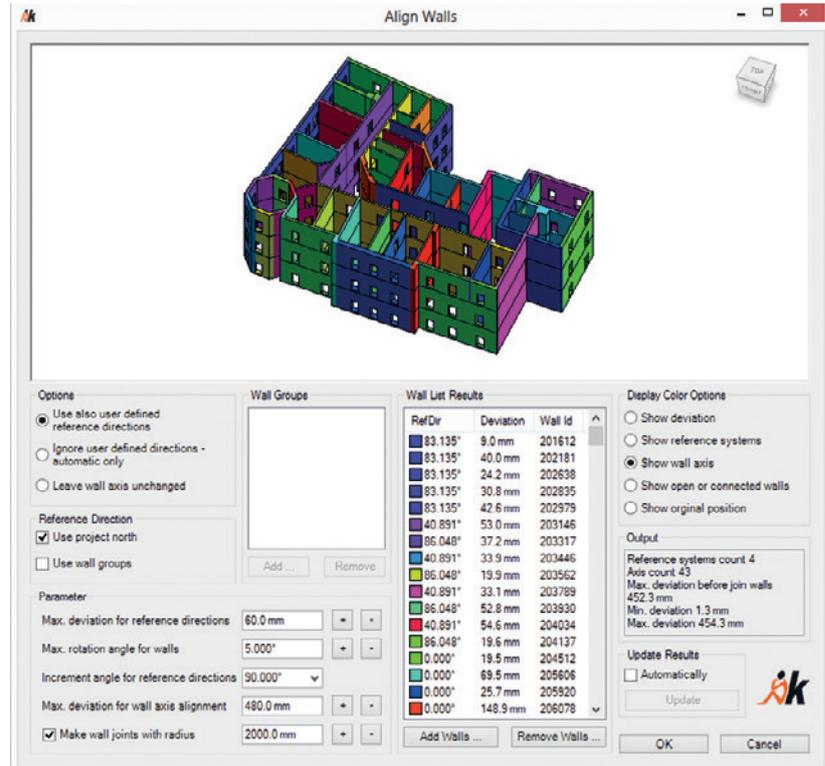


Abb. 10 | „Wände Ausrichten“ – eingefärbte Wandachsen eines Gebäudes mit drei Etagen

## 5 AUSBLICK

Die im Beitrag vorgestellte Methode des „Fittens“ von Wänden soll auf die Bestimmung von weiteren BIM-Bauteiltypen angewendet werden. Zu nennen sind an dieser Stelle die Unterstützung bei der Erkennung von Fenstern, Türen und Öffnungen sowie Trägern und Rohren. Ebenso soll das Ausrichten, Fluchten und Verschneiden auf weitere BIM-Bauteiltypen ausgeweitet werden. Die vorgestellten halbautomatischen Workflows mit ihren Algorithmen ermöglichen ein effizientes Arbeiten. Die Anwendung von Verfahren der Mustererkennung in Punktwolken hat in diesem Bereich ein großes Potenzial.

## LITERATUR

Altman, N. S. (1992): An introduction to kernel and nearest-neighbor non-parametric regression. In: The American Statistician, 46 (1992)3, 175–185.

Braunes, J. (2014): Von der Punktwolke zum Building Information Model (BIM) – Konsequenzen und Wege für die Bestandserfassung. In: VDVmagazin (2014)2, 132–136.

Bringmann, O. (2010): Vom Bauaufmaß zum konsistenten Bauwerksmodell. Nutzerorientierte Bausanierung. Verlag der Bauhaus-Universität Weimar, Weimar, 21–26.

Bringmann, O. (2012): Gekoppelte Objekte in Punktwolken erkennen. In: Luhmann, Th.; Müller, Ch. (Hrsg.): Photogrammetrie – Laserscanning – optische 3D-Messechnik. Beiträge der Oldenburger 3D-Tage 2012. Wichmann, Berlin/Offenbach, 165–172.

Dipl.-Ing. Christian Tonn

FARO 3D SOFTWARE GMBH

Tiergartenstr. 79 | 01219 Dresden  
christian.tonn@faro-europe.com



Dr. Oliver Bringmann

FARO 3D SOFTWARE GMBH

Tiergartenstr. 79 | 01219 Dresden  
oliver.bringmann@faro-europe.com

